

# MPST: A Corpus of Movie Plot Synopses with Tags

Sudipta Kar, Suraj Maharjan, A. Pastor López-Monroy and Thamar Solorio

Department of Computer Science  
University of Houston  
Houston, TX 77204-3010  
{skar3, smaharjan2, alopezmonroy, tsolorio}@uh.edu

## Abstract

Social tagging of movies reveals a wide range of heterogeneous information about movies, like the genre, plot structure, soundtracks, metadata, visual and emotional experiences. Such information can be valuable in building automatic systems to create tags for movies. Automatic tagging systems can help recommendation engines to improve the retrieval of similar movies as well as help viewers to know what to expect from a movie in advance. In this paper, we set out to the task of collecting a corpus of movie plot synopses and tags. We describe a methodology that enabled us to build a fine-grained set of around 70 tags exposing heterogeneous characteristics of movie plots and the multi-label associations of these tags with some 14K movie plot synopses. We investigate how these tags correlate with movies and the flow of emotions throughout different types of movies. Finally, we use this corpus to explore the feasibility of inferring tags from plot synopses. We expect the corpus will be useful in other tasks where analysis of narratives is relevant.

**Keywords:** Tag generation for movies, Movie plot analysis, Multi-label dataset, Narrative texts

## 1. Introduction

Folksonomy (Vander Wal, 2005), also known as collaborative tagging or social tagging, is a popular way to gather community feedback about online items in the form of tags. User-generated tags in recommendation systems like IMDb<sup>1</sup> and MovieLens<sup>2</sup> provide different types of summarized attributes of movies. These tags are effective search keywords, are also useful for discovering social interests, and improving recommendation performance (Lambiotte and Ausloos, 2006; Szomszor et al., 2007; Li et al., 2008; Borne, 2013). In this regard, an interesting research question is: *Can we learn to predict tags for a movie from its written plot synopsis?* This question enables an enormous potential to understand the properties of plot synopses that correlate with the tags. For instance, a movie can be tagged with *fantasy*, *murder*, and *insanity*, that represent different summarized attributes of the movie. The inference of multiple tags by analyzing the written plot synopsis of movies can benefit the recommendation engines. In addition, the consumers would have a useful set of tags representing the plot of a movie. Notwithstanding the usefulness of tags, its proper use in computational methods is challenging as the tag spaces are noisy and redundant (Katakis et al., 2008). Noise and redundancy issues arise because of differences in user perspectives and use of semantically similar tags. For example, the *MovieLens 20M dataset* (Harper and Konstan, 2016), which provides tag assignments between  $\approx 27K$  movies and  $\approx 1,100$  unique tags also suffers from these problems. Thus, a fine-grained tagset and their assignment to movie plots can help to overcome these obstacles.

In this work, (i) we present the MPST corpus that contains plot synopses of 14,828 movies and their associations with a set of 71 fine-grained tags; where each movie is tagged with one or more tags. (ii) We discuss the expected proper-

	<b>A Nightmare on Elm Street 3: Dream Warriors</b> <b>Tags:</b> <i>fantasy, murder, cult, violence, horror, insanity</i>
	<b>50 First Dates</b> <b>Tags:</b> <i>comedy, prank, entertaining, romantic, flashback</i>

Table 1: Examples of tag assignments to movies from the corpus.

ties of this tagset and present the methodology we followed to create such tagset from multiple noisy tag spaces (Section 2.). We also present the process of mapping these tags to a set of movies and collecting the plot synopses for these movies. (iii) We analyze the correlations between the tags and track the flow of emotions throughout the plot synopses to investigate if the associations between tags and movies fit with what we expect in the real world (Section 3.). We also try to estimate the possible difficulty level of a multi-label classification approach to predict tags from the plot synopses. (iv) Finally, we create a benchmark system to predict tags using a set of traditional linguistic features extracted from plot synopses. To the best of our knowledge, this is the first corpus that provides multi-label associations between written plot synopses of movies and a fine-grained tagset. The corpus is freely available to download<sup>3</sup>.

## 2. Creating the Movie Plot Synopses with Tags (MPST) Corpus

There are several datasets that provide plots or scripts of movies. Since their utilization in this work was difficult, we created a fine-grained tagset first and collected the synopses by ourselves. For example, MM-IMDb (Arevalo et al., 2017) provides plot summaries, posters, and metadata of  $\approx 25K$  movies collected from IMDb. But these plot summaries are very short to capture different attributes of

<sup>1</sup><http://www.imdb.com>

<sup>2</sup><https://www.movielens.org>

<sup>3</sup><http://ritual.uh.edu/mpst-2018>



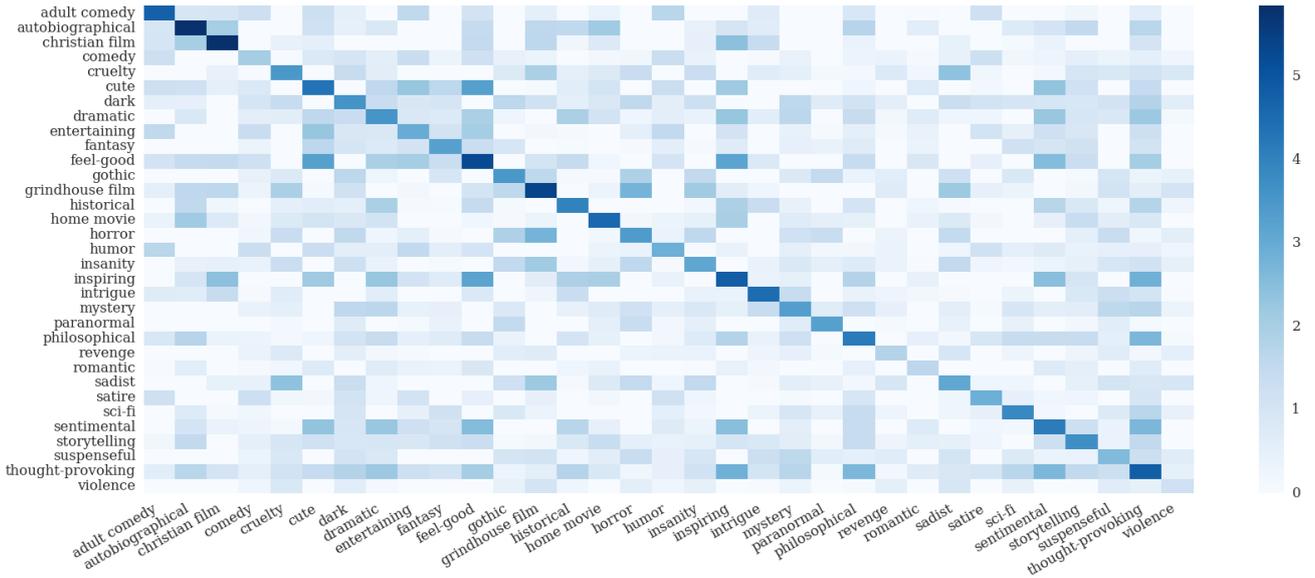


Figure 3: Heatmap of Positive Pointwise Mutual Information (PPMI) between the tags. Dark blue squares represent high PPMI, and white squares represent low PPMI.

Total plot synopses	14,828
Total tags	71
Average tags per movie	2.98
Median value of tags per movie	2
STD of tags for a movie	2.60
Lowest number of tags for a movie	1
Highest number of tags for a movie	25
Average sentences per synopsis	43.59
Median value of sentences per synopsis	32
STD of sentences per synopsis	47.5
Highest number of sentences in a synopsis	1,434
Lowest number of sentences in a synopsis	10
Average words per synopsis	986.47
Median value of words per synopsis	728
STD of words per synopsis	966.16
Highest number of words in a synopsis	13,576
Lowest number of words in a synopsis	72

Table 2: Brief statistics of the MPST corpus.

### 3. Data Statistics

Table 2 shows that the distribution of the number of tags assigned to movies, number of sentences, and number of words per movie are skewed. Most of the synopses are small in terms of the number of sentences, although the corpus contains some really large synopses with more than 1K sentences. Around half of the synopses have less than 33 sentences. A similar pattern is noticeable for the average number of tags assigned to the movies. Some movies have a large number of tags, but most of the movies are tagged with one or two tags only. *Murder*, *violence*, *flash-back*, and *romantic* are the most frequent four tags in the corpus that are assigned to 5,732; 4,426; 2,937 and 2,906 movies respectively. Least frequent tags like *non-fiction*, *christian film*, *autobiographical*, and *suicidal* are assigned to less than 55 movies each.

#### 3.1. Multi-label Statistics

*Label cardinality* (LC) and *label density* (LD) are two statistics that can influence the performance of multi-label learning methods (Tsoumakos and Katakis, 2006; Tsoumakos et al., 2010). Label cardinality is the average number of labels per example in the dataset as defined by Equation 1.

$$LC(D) = \frac{1}{|D|} \sum_{i=1}^{|D|} |Y_i| \quad (1)$$

Here,  $|D|$  is the number of examples in dataset  $D$  and  $Y_i$  is number of labels for the  $i^{th}$  example. Label density is the average number of labels per example in the dataset divided by the total number of labels, as defined by Equation 2.

$$LD(D) = \frac{1}{|D|} \sum_{i=1}^{|D|} \frac{|Y_i|}{|L|} \quad (2)$$

Here,  $|L|$  is the total number of labels in the dataset. Bernardini et al. (2014) analyzed the effects of cardinality and density on multiple datasets. They showed that, for two datasets with similar cardinalities, learning is harder for the one with lower density. And if the density is similar, learning is harder for the one with higher cardinality. For example, learning performance was better for the *Genbase* dataset (LC: 1.252, LD: 0.046) as compared to the *Medical* dataset (LC: 1.245, LD: 0.028), where they had similar cardinalities but the *Medical* dataset was less dense. On the other hand, performance was better for the *Emotions* dataset (LC: 1.869, LD: 0.311) as compared to the *Yeast* dataset (LC: 4.237, LD: 0.303), where they had similar density but cardinality of the *Yeast* dataset was higher. The label cardinality and label density of our dataset are 2.98 and 0.042, respectively. Based on the mentioned experiments, we suspect that a traditional multi-label classification approach for this dataset will be a challenge that opens the scope for exploring more scalable approaches.

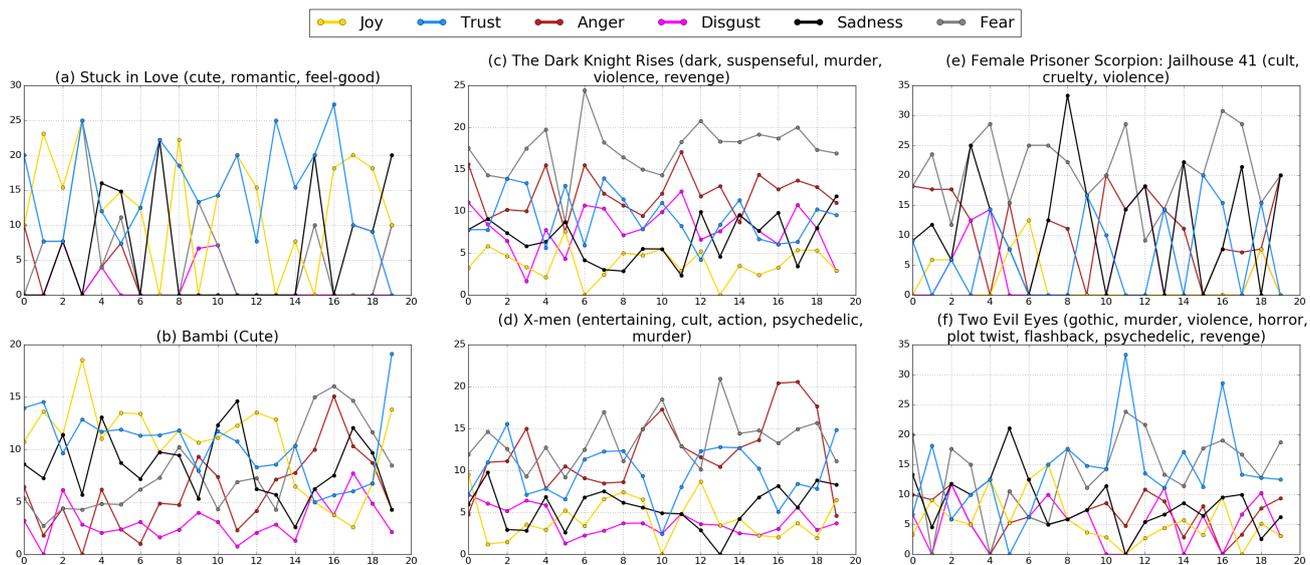


Figure 4: Tracking flow of emotions in the synopses of six movies. Each synopsis was divided into equally sized 20 segments based on the words and percentage of the emotions for each segment were calculated using NRC emotion lexicons. The y axis represents the percentage of emotions in each segment; whereas, the x axis represents the segments.

### 3.2. Correlation between Tags

To find out significant correlations in the tagset, we compute the Positive Pointwise Mutual Information (PPMI) between the tags, which is a modification over the standard PMI (Church and Hanks, 1990; Dagan et al., 1993; Niwa and Nitta, 1994). PPMI between two tags  $t_1$  and  $t_2$  is computed by the following equation:

$$PPMI(t_1; t_2) \equiv \max(\log_2 \frac{P(t_1, t_2)}{P(t_1)P(t_2)}, 0) \quad (3)$$

where,  $P(t_1, t_2)$  is the probability of tags  $t_1$  and  $t_2$  occurring together and  $P(t_1)$  and  $P(t_2)$  are the probabilities of tag  $t_1$  and  $t_2$ , respectively. Figure 3 shows the heatmap correlation of PPMI values between a subset of tags. The figure shows interesting relations between the tags and supports our understanding of the real world scenario.

High PPMI scores show that *cute*, *entertaining*, *dramatic*, and *sentimental* movies can evoke *feel-good* mood, whereas lower PPMI scores between *feel-good* and *sadist*, *cruelty*, *insanity*, and *violence* suggest that these movies usually create a different type of impression on people. Also note that, these movies have stronger relations with *horror*, *cruelty*, and *darkness* which make them difficult to create the *feel-good* experience. It suggest that people tend to get *inspiration* from *dramatic*, *thought-provoking*, *historical*, and *home movies*. *Christian films* and *science fictions* are also good sources of *inspiration*. *Grind-house*, *Christian*, and *non-fiction* films do not usually have *romantic* elements. *Romantic* movies are usually *cute* and *sentimental*. *Autobiographical* movies usually have *story-telling* style and they are *thought-provoking* and *philosophical*. These relations, in fact, show that the movie tags within our corpus seem to portray a reasonable view of movie types based on our understanding of possible impressions from different types of movies.

### 3.3. Emotion Flow in the Synopses

NRC Emotion Lexicons (Mohammad and Turney, 2010) have been shown effective to capture the flow of emotions in narrative stories (Mohammad, 2011). It is a list of 14,182 words<sup>5</sup> and their binary associations with eight types of elementary emotions from the Hourglass of Emotions model (Cambria et al., 2012) (*anger*, *anticipation*, *joy*, *trust*, *disgust*, *sadness*, *surprise*, and *fear*) with polarity.

In Figure 4, we try to inspect how the flows of emotions look like in different types of plots. The reason behind this investigation is to get a shallow idea about the potential feasibility of the collected plot synopses to predict tags. As general users have written the collected plot synopses and created the tags for movies on the web, there is always a possibility to have noise in the data. For example, in a real world scenario we will expect that horror movies will contain fear and sadness. On the other hand, comedy or funny movies will be filled with happiness.

In the figure we can observe that, emotions like *joy* and *trust* are dominant over *disgust* and *anger* in *cute*, *feel-good*, and *romantic* movie's plots (a, b). We can observe sudden spikes in *sadness* in segment 4. The animated movie *Bambi (1942)* shows an interesting flow of different types of emotions. The dominance of *joy* and *trust* suddenly gets low at segment 14 and gets high again at segment 18, where *fear*, *sadness*, and *anger* get high at segment 14. It is quite self-explanatory that the plot are mixtures of positive and negative emotions where the lead characters go through difficult situations, fight enemies and face a happy ending (spike in *joy* and *trust* at the end) after climax scenes where enemies get defeated. The final segments of (b) indicate happy endings, but the rise of *sadness* and *fear* in (a) indicates that *Stuck in Love (2012)* does not have a happy ending.

We observe the opposite scenarios in cases of *violent*, *dark*, *gothic*, and *suspenseful* movies (c, d, e, and f) where

<sup>5</sup>Version 0.92

*fear*, *anger*, and *sadness* dominate over *joy* and *trust*. The dominance of *anger* and *fear* is a good indicator of a movie having *action*, *violence*, and *suspense*. *Female Prisoner Scorpion: Jailhouse 41 (1972)* (e), has dominance of *fear*, *sadness*, and *anger* throughout the whole movie, and it is easy to guess that this movie has *violence* and *cruelty* portrayed through the lead characters. The flow of *joy*, *trust*, *sadness*, and *fear* alters at the middle of the movie *Two Evil Eyes (1990)* (f). Maybe it is the reason why people tagged it with *plot twist*. These observations give evidence of the connection between the flow of emotion in the plot synopses and the experience people can have from the movies, and they also match with what we expected.

## 4. A Machine Learning Approach for Predicting Tags using Plot Synopses

In this section, we will discuss about some preliminary experiments we conduct with the corpus for predicting tags for movies. We approach the task of predicting tags for movies as a multi-label classification problem and use various traditional linguistic features.

### 4.1. Hand-crafted Features

**Lexical:** We extract word  $n$ -grams ( $n=1,2,3$ ), character  $n$ -grams ( $n=3,4$ ) and two skip  $n$ -grams ( $n=2,3$ ) from the plot synopses as they are strong lexical representations. We use term frequency-inverse document frequency (TF-IDF) as the weighting scheme.

**Sentiments and Emotions:** Sentiments are inherent part of stories and one of the key elements that determine the possible experiences found from a story. For example, *depressive* stories are expected to be full of *sadness*, *anger*, *disgust* and negativity, whereas a funny movie is possibly full of *joy* and *surprise*. In this work, we employ two approaches to capture sentiment related features.

- **Bag of Concepts:** As concept-level information have showed effectiveness in sentiment analysis (Cambria, 2013), we extract around 10K unique concepts from the plot synopses using the Sentic Concept parser<sup>6</sup>. It breaks sentences into verb and noun clauses and extracts concepts from them using Parts of Speech (POS) based bigram rules (Rajagopal et al., 2013).
- **Affective Dimensions Scores:** The hourglass of emotions model (Cambria et al., 2012) categorized human emotions into four affective dimensions (*attention*, *sensitivity*, *aptitude* and *pleasantness*) starting from the study on human emotions by Plutchik (2001). Each of these affective dimensions is represented by six different activation levels called ‘sentic levels’. These make up to 24 distinct labels called ‘elementary emotions’ that represent the total emotional state of the human mind. SenticNet 4.0 (Cambria et al., 2016) knowledge base consists of 50,000 common-sense concepts with their semantics, polarity value and scores for the basic four affective dimensions. We

used this knowledge base to compute average polarity, attention, sensitivity, aptitude, and pleasantness for the synopses.

We divide the plot synopses into three equal chunks based on words and extracted these two sentiment features for each chunk. We will discuss more about chunk-based sentiment representation later.

**Semantic Frames:** Semantic role labeling is a useful technique to assign abstract roles to the arguments of predicates or verbs of sentences. We use SEMAFOR<sup>7</sup> frame-semantic parser to parse the frame-semantic structure using the FrameNet (Baker et al., 1998) frames. For each synopsis, we use the bag of frames representation weighted by normalized frequency as feature.

**Word Embeddings:** Word embeddings have been shown effectiveness in text classification problems by capturing semantic information. Hence, in order to capture the semantic representation of the plots, we average the word vectors of every word in the plot. We use the publicly available FastText pre-trained word embeddings<sup>8</sup>.

**Agent Verbs and Patient Verbs:** Actions done and received by the characters can help to identify attributes of plots. For example, if the characters of a movie *kill*, *take revenge*, *shoot*, *smuggle*, *chase*; we can expect violence, murder, action from that story. We use the agent and patient verbs found in synopses to capture the actions. In this regard, we use Stanford CoreNLP library to parse the dependencies of the synopses. Then we extract the agent verbs (using *nsubj* or *agent* dependencies) and the patient verbs (using *dobj*, *nsubjpass*, *iobj* dependencies) as described in Bamman et al. (2013). We group these verbs into 500 clusters using the pre-trained word embeddings with the K-means clustering algorithm to reduce noise. We use the distribution of these clusters of the agent verbs and patient verbs over the synopses. We experimented with different values of K (K=100, 500, 1000, 1500), and 500 clusters helped to achieve better results.

### 4.2. Experimental Setup

Section 3. shows that the distribution of the number tags assigned to per movies is skewed. The average number of tags per movie is approximately three. We thus begin by experimenting with predicting a fixed number of three tags for each movie. Moreover, to get more detailed idea about movies, we create another set of five tags by predicting two additional tags.

We use random stratified split to divide the data into 80:20 train to test ratio<sup>9</sup>. We use the One-versus-Rest approach to predict multiple tags for an instance. We experiment with logistic regression as the base classifier. We run five-fold cross-validation on the training data to evaluate different features and combinations. We tune the regularization parameter ( $C$ ) using grid search technique over the best feature combination that includes all of the extracted features. We use the best parameter value ( $C=0.1$ ) for training

<sup>7</sup><http://www.cs.cmu.edu/~ark/SEMAFOR>

<sup>8</sup><https://github.com/facebookresearch/fastText/blob/master/pretrained-vectors.md>

<sup>9</sup>Train-test partition information is available with the dataset.

<sup>6</sup><https://github.com/SenticNet/concept-parser>

a model with all the training data and used that model for predicting tags for the test data.

**Majority and Random Baseline:** We define majority and random baselines to compare the performance of our proposed model in the task of predicting tags for movies. The majority baseline method assigns the most frequent three or five tags to all the movies. We chose three tags per movie as this is the average number of tags per movie in the dataset. Similarly, the random baseline assigns at random three or five tags to each movie.

**Evaluation Metrics:** Wu and Zhou (2016) illustrate the complications in evaluating multi-label classifiers by an example of determining the significance of mistakes for the following cases: *one instance with three incorrect labels vs. three instances each with one incorrect label*. It is complicated to tell which of these mistakes is more serious. Due to such complications, several evaluation methodologies have been proposed for this type of tasks (Tsoumakas and Katakis, 2006; Wu and Zhou, 2016). For example, *hmming loss*, *average precision*, *ranking loss*, *one-error*, *coverage*, (Schapire and Singer, 2000; Fürnkranz et al., 2008), *micro* and *macro* averaged versions of *F1* and *AUC* score (Tsoumakas et al., 2010; Tsoumakas et al., 2011; Lipton et al., 2015).

Another complication arises when the label distribution is sparse in a dataset. Less frequent tags could be under-represented by models, but an ideal model should be able to discriminate among all the possible labels. Such an issue is very common in problems like image annotation, and existing works use *mean per label recall* and *labels with recall > 0* to measure the effectiveness of models in learning individual labels (Lavrenko et al., 2003; Feng et al., 2004; Carneiro et al., 2007; Wang et al., 2009). Here, we use two similar metrics: *tag recall* (TR) and *tags learned* (TL), along with traditional micro-F1 metric. Tag recall computes the average recall per tag and defined by the following equation.

$$TR = \frac{\sum_{i=1}^{|T|} |R_i|}{|T|} \quad (4)$$

Here,  $|T|$  is the size of tagset in the corpus, and  $R_i$  is recall of  $i^{th}$  tag. Tags learned (TL) computes how many unique tags are being predicted by the system for the test data. These evaluation metrics will help us to investigate how well and how many distinct tags are being learned by the models. We evaluate the models using these three metrics in two settings. One is selecting the top three tags and another is selecting the top five tags.

### 4.3. Results and Analysis

Table 3 shows the performance of the hand-crafted features for predicting tags for movies. All the features beat the baselines in terms of micro-F1 and tag recall (TR). But another significant criterion to evaluate the performances is the number of unique tags predicted by the models, which is measured by the tags learned (TL) metric. We prefer such a model that is capable of creating diverse tagsets by capturing varieties of attributes of movies with reasonable accuracy. For instance, the random baseline used all of the tags in the dataset to assign to the movies but its accuracy

	Top 3			Top 5		
	F1	TR	TL	F1	TR	TL
Baseline: Most Frequent	29.7	4.225	3	31.5	7.042	5
Baseline: Random	4.20	4.328	71	5.40	7.281	71
Unigram (U)	37.6	7.883	22.6	37.1	11.945	27.4
Bigram (B)	36.5	7.216	19.6	36.1	10.808	24.8
Trigram (T)	31.3	5.204	15.4	32.4	8.461	21
Char 3-gram (C3)	37.0	7.419	22	36.6	11.264	27.4
Char 4-gram (C4)	37.7	7.799	22.6	37.0	11.582	27.2
2 skip 2 gram (2S2)	34.2	6.289	19.4	34.5	9.875	25.2
2 skip 3 gram (2S3)	30.8	4.951	12.8	32.1	8.109	18.2
Bag of Concepts (BoC)	35.7	7.984	29	35.9	12.473	34.8
Concepts Scores (CS)	31.1	4.662	7.8	32.4	7.512	8.2
Word Embeddings	36.8	6.744	13.2	36.1	10.074	17.8
Semantic Frame	33.4	5.551	13.4	33.9	8.394	15.2
Agent Verbs	32.9	5.050	7.2	33.2	7.714	8
Patient Verbs	33.1	5.134	7.4	33.5	7.843	8
U+B+T	37.2	8.732	30	36.8	13.576	36.8
C3+C4	<b>37.8</b>	8.662	28.8	<b>37.4</b>	13.395	33.6
U+B+T+C3+C4	37.1	9.991	36.8	36.8	15.871	45.8
All lexical	36.7	10.046	37.6	36.5	15.838	46.4
BoC + CS	35.7	8.165	29.4	36.0	12.754	35.4
<b>All features</b>	36.9	<b>10.364</b>	39.6	36.8	<b>16.271</b>	47.8

Table 3: Performance of the hand-crafted features using 5-fold cross-validation on the training data. We use three metrics (*F1*: micro averaged *F1*, *TR*: tag recall, and *TL*: tags learned) to evaluate the features.

	Top 3			Top 5		
	F1	TR	TL	F1	TR	TL
Baseline: Most Frequent	29.7	4.23	3	28.4	14.08	5
Baseline: Random	4.20	4.21	71	6.36	15.04	71
<b>System</b>	<b>37.3</b>	<b>10.52</b>	47	<b>37.3</b>	<b>16.77</b>	52

Table 4: Results achieved on the test data using the best feature combination (all features) with tuned regularization parameter  $C$ .

is very poor. On the other hand, the majority baseline has better accuracy but it does not have diversity in the tagset. We can see that most of the individual features achieved almost similar micro-F1 scores, but they demonstrate difference in effectiveness to create diversity in predicted tags. Feature combinations seem to improve in TR and TL, but micro-F1 scores are almost similar to the individual features.

The lexical features show better performance compared to other features. Bag of concepts (BoC) shows similarity in performance. Combination of all lexical features demonstrates effectiveness in capturing a wide range of attributes of movies from the synopses, which is reflected by the better TR and TL scores.

We present the results achieved on the test data in Table 4. Although the result is similar to the result we got with all features during cross-validation, number of predicted unique tags is higher in the test set. This result could be used as a baseline system to compare other methods developed in future as it uses several traditional linguistic features combination to predict tags.

**Chunk-based Sentiment Representation:** Narratives have patterns in ups and downs of sentiments (Vonnegut, 1981). Reagan et al. (2016) showed that the pattern of changes in sentiments is significant for consumer experiences that results in success of stories. To capture such changes, we experiment with chunk-based sentiments

Chunks	Top 3			Top 5		
	F1	TR	TL	F1	TR	TL
1	35.2	6.550	18.2	35.1	9.928	23.4
2	35.0	7.031	23.0	35.2	10.68	26.8
3	<b>35.7</b>	8.165	29.4	<b>36.0</b>	<b>12.754</b>	35.4
4	35.1	8.153	30.6	35.4	12.723	<b>36.8</b>
5	34.8	<b>8.185</b>	30.4	35.1	12.553	<b>36.8</b>
6	34.3	7.976	<b>31.2</b>	34.9	12.725	36.0

Table 5: Experimental results obtained by 5-fold cross-validation using chunk-based sentiment representations. Chunk-based sentiment features were combined with the other features described in Section 4.1.

and emotions representation. We divide the plot synopses into equally sized  $n$  chunks based on the word tokens and extract the sentiment and emotion features for each chunk. Then we run 5-fold cross validation on the training data to observe the effect of chunk-based sentiments and emotions representation. We report the results in Table 5. Results show that dividing synopses into multiple chunks and using sentiment and emotion features from each chunk improves the performance of tag prediction. Although we observe noticeable improvements up to three chunks, TL remains similar where micro-F1 scores start to drop when we use more than three chunks. We suspect that higher number of chunks create sparseness in the representation of sentiments and emotions that hurts the performance. So we use sentiments and emotions features using three chunks in further experiments. As the chunk-based representation shows improvement in results, we plan to work capturing the flow of sentiments throughout the plots more efficiently in future work.

## 5. Conclusion

We have presented a new corpus of  $\approx 70$  fine-grained tags and their associations with  $\approx 14K$  plot synopses of movies. In order to create the tagset, we tackled the challenge of extracting tags related to movie plots from noisy and redundant tag spaces created by user communities in MovieLens and IMDb. In this regard, we describe the methodology for creating the fine-grained tagset and mapping the tags to the plot synopses.

We present an analysis, where we try to find out the correlations between tags. These correlations seem to portray a reasonable set of movie types based on what we expect from certain types of movies in the real world. We also try to analyze the structure of some plots by tracking the flow of emotions throughout the synopses, where we observed that movies with similar tag groups seem to have similarities in the flow of emotions throughout the plots.

Finally, we create a benchmark system to predict tags from the synopses using a set of hand-crafted linguistic features. This dataset will be helpful to analyze and understand the linguistic characteristics of plot synopses of movies, which will in turn help to model certain types of abstractions as tags. For example, what type of events, word choices, character personas, relationships between characters, and plot structure make a movie *mysterious* or *suspenseful* or

*paranormal*? Such investigations can help the research community to better exploit high-level information from narrative texts, and also help to build automatic systems to create tags for movies. The generation of tags from movie plots or narrative texts could also be a significant step towards solving the problem of automatic movie profile generation. Methodologies designed using the MPST corpus could also be used to analyze narrative texts from other domains, such as books and storyline of video games.

## Acknowledgements

We would like to thank the National Science Foundation for partially funding this work under award 1462141. We are also grateful to Prasha Shrestha, Giovanni Molina, Deepthi Mave, and Gustavo Aguilar for reviewing and providing valuable feedback during the process of creating tag clusters.

## 6. Bibliographical References

- Arevalo, J., Solorio, T., y Gómez, M. M., and González, F. A. (2017). Gated multimodal units for information fusion. In *5th International Conference on Learning Representations (ICLR) 2017 -Workshop Track*.
- Baker, C. F., Fillmore, C. J., and Lowe, J. B. (1998). The berkeley framenet project. In *Proceedings of the 17th international conference on Computational linguistics-Volume 1*, pages 86–90. Association for Computational Linguistics.
- Bamman, D., O’Connor, B., and Smith, N. A. (2013). Learning latent personas of film characters. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 352–361, Sofia, Bulgaria, August. Association for Computational Linguistics.
- Bernardini, F. C., da Silva, R. B., Rodovalho, R. M., and Meza, E. B. M. (2014). Cardinality and density measures and their influence to multi-label learning methods. *Dens*, 1:1.
- Borne, K. (2013). Collaborative annotation for scientific data discovery and reuse. *Bulletin of the American Society for Information Science and Technology*, 39(4):44–45.
- Cambria, E., Livingstone, A., and Hussain, A. (2012). The hourglass of emotions. In *Proceedings of the 2011 International Conference on Cognitive Behavioural Systems, COST’11*, pages 144–157, Berlin, Heidelberg. Springer-Verlag.
- Cambria, E., Poria, S., Bajpai, R., and Schuller, B. (2016). Senticnet 4: A semantic resource for sentiment analysis based on conceptual primitives. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 2666–2677, Osaka, Japan, December. The COLING 2016 Organizing Committee.
- Cambria, E. (2013). An introduction to concept-level sentiment analysis. In *Mexican International Conference on Artificial Intelligence*, pages 478–483. Springer.
- Carneiro, G., Chan, A. B., Moreno, P. J., and Vasconcelos, N. (2007). Supervised learning of semantic classes

- for image annotation and retrieval. *IEEE Trans. Pattern Anal. Mach. Intell.*, 29(3):394–410, March.
- Church, K. W. and Hanks, P. (1990). Word association norms, mutual information, and lexicography. *Comput. Linguist.*, 16(1):22–29, March.
- Dagan, I., Marcus, S., and Markovitch, S. (1993). Contextual word similarity and estimation from sparse data. In *Proceedings of the 31st Annual Meeting on Association for Computational Linguistics*, ACL '93, pages 164–171, Stroudsburg, PA, USA. Association for Computational Linguistics.
- Feng, S. L., Manmatha, R., and Lavrenko, V. (2004). Multiple bernoulli relevance models for image and video annotation. In *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, CVPR'04, pages 1002–1009, Washington, DC, USA. IEEE Computer Society.
- Fürnkranz, J., Hüllermeier, E., LozaMencía, E., and Brinker, K. (2008). Multilabel classification via calibrated label ranking. *Machine Learning*, 73(2):133–153, Nov.
- Gorinski, P. J. and Lapata, M. (2015). Movie script summarization as graph-based scene extraction. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1066–1076, Denver, Colorado, May–June. Association for Computational Linguistics.
- Katakis, I., Tsoumakas, G., and Vlahavas, I. (2008). Multi-label text classification for automated tag suggestion. In *Proceedings of the ECML/PKDD 2008 Discovery Challenge*.
- Lambiotte, R. and Ausloos, M., (2006). *Collaborative Tagging as a Tripartite Network*, pages 1114–1117. Springer Berlin Heidelberg, Berlin, Heidelberg.
- Lavrenko, V., Manmatha, R., and Jeon, J. (2003). A model for learning the semantics of pictures. In *Proceedings of the 16th International Conference on Neural Information Processing Systems*, NIPS'03, pages 553–560, Cambridge, MA, USA. MIT Press.
- Li, X., Guo, L., and Zhao, Y. E. (2008). Tag-based social interest discovery. In *Proceedings of the 17th International Conference on World Wide Web*, WWW '08, pages 675–684, New York, NY, USA. ACM.
- Lipton, Z. C., Kale, D. C., Elkan, C., and Wetzell, R. C. (2015). Learning to diagnose with LSTM recurrent neural networks. *CoRR*, abs/1511.03677.
- Mohammad, S. (2011). From once upon a time to happily ever after: Tracking emotions in novels and fairy tales. In *Proceedings of the 5th ACL-HLT Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities*, LaTeCH '11, pages 105–114, Stroudsburg, PA, USA. Association for Computational Linguistics.
- Niwa, Y. and Nitta, Y. (1994). Co-occurrence vectors from corpora vs. distance vectors from dictionaries. In *Proceedings of the 15th Conference on Computational Linguistics - Volume 1*, COLING '94, pages 304–309, Stroudsburg, PA, USA. Association for Computational Linguistics.
- Plutchik, R. (2001). The nature of emotions human emotions have deep evolutionary roots, a fact that may explain their complexity and provide tools for clinical practice. *American scientist*, 89(4):344–350.
- Rajagopal, D., Cambria, E., Olsher, D., and Kwok, K. (2013). A graph-based approach to commonsense concept extraction and semantic similarity detection. In *Proceedings of the 22nd International Conference on World Wide Web*, pages 565–570. ACM.
- Reagan, A. J., Mitchell, L., Kiley, D., Danforth, C. M., and Dodds, P. S. (2016). The emotional arcs of stories are dominated by six basic shapes. *CoRR*, abs/1606.07772.
- Schapire, R. E. and Singer, Y. (2000). Boostexter: A boosting-based system for text categorization. *Machine learning*, 39(2-3):135–168.
- Szomszor, M., Cattuto, C., Alani, H., O'Hara, K., Baldassarri, A., Loreto, V., and Servedio, V. D. (2007). Folksonomies, the semantic web, and movie recommendation.
- Tsoumakas, G. and Katakis, I. (2006). Multi-label classification: An overview. *Dept. of Informatics, Aristotle University of Thessaloniki, Greece*.
- Tsoumakas, G., Katakis, I., and Vlahavas, I. (2010). Mining multi-label data. In *Data Mining and Knowledge Discovery Handbook*, pages 667–685.
- Tsoumakas, G., Katakis, I., and Vlahavas, I. (2011). Random k-labelsets for multilabel classification. *IEEE Transactions on Knowledge and Data Engineering*, 23(7):1079–1089.
- Vander Wal, T. (2005). Folksonomy definition and wikipedia. *vanderwal.net*.
- Vonnegut, K. (1981). Palm sunday: An autobiographical collage.
- Wang, C., Yan, S., Zhang, L., and Zhang, H.-J. (2009). Multi-label sparse coding for automatic image annotation. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1643–1650, June.
- Wu, X. and Zhou, Z. (2016). A unified view of multi-label performance measures. *CoRR*, abs/1609.00288.

## 7. Language Resource References

- Harper, F Maxwell and Konstan, Joseph A. (2016). *The movielens datasets: History and context*. ACM.
- Mohammad, Saif and Turney, Peter D. (2010). *Emotions Evoked by Common Words and Phrases: Using Mechanical Turk to Create an Emotion Lexicon*. Association for Computational Linguistics.